

TECHNICKÁ UNIVERZITA V LIBERCI
Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: B26462 – Informační technologie

Studijní obor: 1802R007 – Informační technologie

Evidence přidělované práce

Task manager

Bakalářská práce

Autor: **Jan Wanke**

Vedoucí práce: Ing. Alena Gregová

Konzultant: Petr Mojžíš, Dis.

V Liberci 2. 1. 2013

Zadání práce

Prohlášení

Byl jsem seznámen s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/200 Sb. o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím bakalářské práce a konzultantem.

Datum

2. 1. 2013

Podpis

Poděkování

Chtěl bych poděkovat především vedoucí této bakalářské práce Ing. Aleně Gregové, za její ochotu a čas věnovaný konzultacím. Dále konzultantovi Petru Mojžíšovi, Dis., který mi ukázal nové praktiky při programování, za jeho vedení a směřování mé práce a hlavně za jeho trpělivost při mých prvních nezdarech. Poděkování také patří mé rodině za jejich podporu při psaní této bakalářské práce, ale i během celého studia. Dále též patří všem mým přátelům a kamarádům, kteří při mně stáli, a se kterými jsem mohl konzultovat případné dotazy. V neposlední řadě patří poděkování i mé přítelkyni, která mi je každodenní oporou a bez které by tato práce pravděpodobně nevznikla.

Předmluva

Bakalářská práce je koncipována jako uživatelská příručka. Nenajdete v ní proto žádné konkrétní kusy zdrojových kódů, žádné konkrétní rozložení tabulek v databázi apod. kromě demonstrativní ukázky třídy, která se stará o vykreslování grafických prvků na stránkách (viz příloha 3). Kvůli bezpečnosti aplikace není možné zdrojové kódy zveřejnit. Aplikace bude využita v komerčním prostředí, a proto by bylo nežádoucí zveřejnění kritických bodů práce. Práce je vytvářena pro nejmenovanou firmu, zastoupenou konzultantem Petrem Mojžíšem, Dis.

V teoretické části bakalářské práce jsou popsány hlavní a základní technologie, které jsou použity pro finální aplikaci a jejich použití je v praktické části.

V praktické části je obsažena samotná uživatelská příručka se základními principy práce s aplikací, její omezení v podobě uživatelských oprávnění pro přístup do jednotlivých sekcí. Návod je doplněn o screenshoty aplikace, vztahujících se k aktuálnímu tématu.

Abstrakt

Bakalářská práce slouží jako uživatelská příručka pro uživatele webové aplikace Task Manager, která byla vytvořena v rámci této práce. Aplikace slouží uživatelům pro koordinaci a správu práce na jejich projektech a dílčích úkolech. Dokáže evidovat čas strávený na projektech, vytvářet statistiky, přidělovat projekty a úkoly zaměstnancům. Pomocí ní lze snadno upravovat zaměstnance, mazat je, přidávat do firmy a celkově koordinovat práci na projektech a úkolech v rámci firmy z pohledu projektového manažera, který si může udělat lepší obrázek o vytížení a rozvržení práce svých kolegů. To vše za pomoci této webové aplikace.

Bakalářská práce je rozdělena na dvě části. Na teoretickou a praktickou.

V teoretické části jsou stručně popsány a vysvětleny technologie použité v aplikaci Task Manager. Každá technologie obsahuje krátkou historii a její popis pro snazší porozumění vysvětlovaným termínům v praktické části.

V praktické části naleznete uživatelskou příručku. Tato práce je koncipována jako manuál pro uživatele, kteří budou používat aplikaci. V celé bakalářské práci nejsou žádné části zdrojových kódů, ani nijak popsány bezpečnostní pilíře. Důvodem je zabezpečení práce proti případným útokům na aplikaci a neulehčování tím tak práce útočníkům. Je zde vysvětlena orientace na stránkách, jednotlivé položky menu a jejich funkcionality. Díky této příručce by se měl uživatel snadněji a s větší jistotou pohybovat po webu a měla by mu pomoci s prvními kroky v aplikaci.

Klíčová slova

Evidence přidělované práce, Task Manager, webová aplikace, uživatelská příručka

Abstract

Bachelor thesis serves as a user manual for users of a web application called Task Manager, that was created as part of this bachelor thesis. The application serves users for co-ordinating and administration of work on their projects and fractional tasks. It can register time spent on projects, make statistics, assessing projects and tasks to employees. With this application you can easily edit employees, delete and join them to companies. Another use is being able to co-ordinate your work on projects and tasks inside your company from the Projects Manager's view, who can better oversee employees's workloads and work patterns of his colleagues. All of these functions work because of this application.

Bachelor thesis is divided into two parts, a theoretical and a practical one.

Technologies used in application are concisely described and explained in theoretical part of this bachelor thesis. Each technology is considering short history and it's description for easily understanding terms are used in the practical part.

You can find the user's manual in the practical part. This thesis is written for users who will use this application. There are no parts of source code either for any descriptions of secure pillars in this thesis. The reason is to secure work against attacks to this application and do not make it easy for attackers. There is an explanation of orientation in the application, individual items of menus and their functions. This should help the manual user to use the application more easily and be more comfortable moving around within it. This will be an essential guide for help with the first steps of the application.

Key words:

Records of allocated work, Task Manager, web application, user's manual

Obsah

Titulní strana	1
Zadání práce.....	Chyba! Záložka není definována.
Prohlášení	3
Poděkování.....	4
Předmluva	4
Abstrakt	5
Abstract.....	6
Obsah.....	7
Teoretická část	9
1. Jazyk JavaScript	10
1.1 Historie jazyka	10
1.2 Vlastnosti jazyka	10
1.3 AJAX.....	11
1.4 Knihovna jQuery.....	11
1.5 Knihovna jQuery UI.....	12
1.5.1 Kategorie knihovny a jejich funkce, elementy, efekty.....	13
1.6 jqPlot.....	15
2. Jazyk PHP	16
2.1 Historie jazyka	16
2.2 Vlastnosti jazyka	17
3. MySQL	18
3.1 Historie MySQL.....	18
3.2 Vlastnosti MySQL	18
3.3 Historie SQL	19
4. Jazyk HTML	19
4.1 Historie jazyka	19
4.2 Vlastnosti jazyka	20
5. CSS	20
5.1 Historie CSS	20
5.2 Vlastnosti CSS	21
Praktická část	22
6. Aplikace TaskManager	23
6.1 Seznámení s aplikací a její historie	23
6.2 První příchod na webové stránky.....	24

6.3	Horní navigace – hlavní menu	25
6.3.1	Sekce „domů“	26
6.3.2	Sekce „projekty“	26
6.3.3	Sekce „úkoly“	30
6.3.4	Sekce „zaměstnanci“	32
6.3.5	Sekce „statistiky“	33
6.3.6	Sekce „budget“	34
6.3.7	Sekce „administrace“	34
6.3.8	Sekce „přihlásit / odhlásit“	37
7.	Závěrem	38
7.1	Problémy při vypracovávání	38
7.2	Bezproblémové části práce	38
7.3	Přínos práce pro autora	38
7.4	Stanovené cíle, splněné a nesplněné úkoly	39
	Seznam použité literatury a odkazů:	40
	Seznam obrázků:	41
	Příloha č. 1	42
	Příloha č. 2	43
	Příloha č. 3	45
	Příloha č. 4	47

Teoretická část

1. Jazyk JavaScript

1.1 Historie jazyka

Jazyk JavaScript byl vydán v roce 1995 jako doplněk k jazykům HTML a Java. Jeho autorem je Brendan Eich. Tento jazyk patří do rodiny jazyků C/C++/Java, se kterými má podobnou syntaxi. JavaScript byl obchodním názvem implementace společnosti Netscape, kde byl poté vyvíjen pod názvem Mocha a později pod názvem LiveScript. [14]

1.2 Vlastnosti jazyka

Jazyk JavaScript je interpretovaný, skriptovací, nikoli kompilovaný, objektově-orientovaný jazyk. Zpracování skriptů probíhá na straně klienta, tedy v prohlížeči. To znamená, že program se spouští až po stažení WWW stránky z internetu. Jazyk je *case sensitive*, záleží tedy na velikosti písmen v kódu, je závislý na prohlížeči. Uživatel si na své straně může v prohlížeči JavaScript vypnout/zakázat, čímž může ovlivnit funkčnost stránek. Datové typy v jazyce JavaScript jsou asociovány s hodnotami, nikoli proměnnými. To znamená, že u proměnné neurčujeme, jaký má datový typ. Pouze proměnnou deklarujeme a případně inicializujeme hodnotu. Do proměnné se později může zapsat jiná hodnota s odlišným datovým typem, než co byl v původní proměnné. Tomu se též říká dynamické přiřazení typů (*dynamic typing*). Dnes se jazyk nejčastěji používá ke kontrole formulářových prvků před odesláním dat na server, animacím a různým efektům prvků jako jsou obrázky, tlačítka atd. [14]

Technologie JavaScript byla použita hlavně ke kontrolám formulářů, tzn. jejich validaci, k ovládání stopek, které měří čas odpracovaný na úkolech, k blokování tlačítek a jejich opětovnému aktivování/deaktivování a dalším dílčím akcím.



Obr. 1: Logo JavaScript

1.3 AJAX

AJAX z anglického *Asynchronous JavaScript and XML* je asynchronní JavaScript a značovací jazyk XML (*Extensible Markup Language*). Jedná se o technologii umožňující měnit obsah webových stránek na základě podnětu od uživatele, např. aktualizace komentářů na stránce, aktualizování položek v e-shopu při změně parametru vyhledávání bez nutnosti znovu načtení stránek. Pokud by uživatel chtěl aktualizovat výpis komentářů, musel by odeslat požadavek na server, ten by jej zpracoval a odeslal zpět výsledek. Stránka by se aktualizovala a vykreslila celá znovu. Při načtení by stránka byla vyscrollována zpět na začátek (v případě delší stránky, která se nevejde celá na obrazovku). Při použití technologie AJAX se odesílá požadavek na server v pozadí a aktualizují se pouze ty prvky, ve kterých došlo ke změně. Zároveň nedochází k nechtěnému efektu odscrollování stránky a pozice na stránce zůstává neměnná. Díky této technologii se mění zátěž serveru, protože není potřeba vykreslovat celý HTML dokument, ale jen prvky, u kterých proběhla změna. [13]

Technologie AJAX byla využita v oblasti statistik, kde byl odeslán požadavek na server a ten zpět vracel data k zobrazení. Další využití našla u formulářů, kde jsou výběry země, kraje, okresu a města. Bylo potřeba, aby se po zvolení země odeslal požadavek na server a do následujícího výběrového okna se vložily pouze kraje, které odpovídají vybrané zemi. Stejný postup byl použit u krajů, okresů a měst.



Obr. 2: Logo AJAX

1.4 Knihovna jQuery

Knihovna jQuery je velmi oblíbenou javascriptovou knihovnou. Základním cílem této knihovny je smazat rozdíly mezi implementacemi jazyka JavaScript v různých webových prohlížečích. Knihovna se nesnaží zavést objektově-orientovaný způsob programování v jazyce JavaScript, ale místo toho použít JavaScript jako jazyk

řízený událostmi. Snaží se o minimalizaci délky kódu a zavádí princip řetězení volání svých metod. Poprvé byla vydána Johnem Resigem v srpnu roku 2006 jako knihovna s otevřeným kódem. Je nabízena pod licencemi MIT (*licence se vznikem na Massachusetts Institute of Technology*) a GPL (*General Public Licence*) [1] [12].

Existuje několik verzí knihovny. Od jQuery 1.0 až po poslední stabilní verzi jQuery 1.7 (aktuální v době psaní této bakalářské práce). Jsou volně ke stažení z oficiálních stránek www.jquery.com.

Knihovna jQuery byla použita ve velkém rozsahu. Díky ní jsou na stránce tabulky, u kterých lze řadit data uvnitř, pracuje s ní plugin simplemodal, který se stará o vykreslování dialogových oken. Dále umožňuje zobrazování a skrývání formulářových prvků v závislosti na předchozích atd.



Obr. 3: Logo jQuery

1.5 Knihovna jQuery UI

Knihovna jQuery UI je také knihovna s otevřeným zdrojovým kódem. Slouží hlavně ke stylování a implementování pokročilejších efektů. Její komponenty se dělí do 4 kategorií:

- interakce (*Interactions*)
- ovládací prvky (*Widgets*)
- efekty (*Effects*)
- pomocné nástroje (*Utilities*)

Samotnou kategorií je jádro (*UI Core*). Knihovna má k dispozici řadu motivů vzhledu, které lze použít. Je zde i možnost vytvořit si vlastní vzhled. Poprvé byla vydána v září roku 2007 a je nabízena pod licencemi MIT (*licence se vznikem na Massachusetts Institute of Technology*) a GPL (*General Public Licence*). Základem byla knihovna jQuery [7] [12].

1.5.1 Kategorie knihovny a jejich funkce, elementy, efekty

Kategorie interakce (*Interactions*) obsahuje tyto funkce:

Draggable – Tato funkce umožňuje přiřadit objektu vlastnost přesouvání. Objekt s touto přiřazenou vlastností je možné libovolně přesouvat na stránce.

Droppable – Pomocí této funkce lze implementovat vlastnost drag and drop, čili uchopení objektu s vlastností drag, jeho následné přesunutí do pozice, kde se nachází objekt s vlastností drop. Příkladem jsou e-shopy a vkládání produktů do košíků.

Resizable – Díky této funkci lze objektu měnit jeho velikost tzn. zvětšovat jej či zmenšovat.

Selectable – Tato funkce umožňuje vybrat buďto jeden nebo i několik elementů.

Sortable – Za pomoci této funkce lze třídit a řadit objekty příp. elementy.

Kategorie ovládacích prvků (*Widgets*) obsahuje tyto elementy:

Accordion – Pomocí toho elementu lze vytvořit prvek resp. blok, který slouží k rozdělení informací do sekcí. Zobrazení vypadá tak, že jsou vidět pouze popisky sekcí a jedna sekce je aktivní. Při změně sekce (po kliknutí na popisek sekce) se nejprve uzavře aktivní sekce a poté se rozbalí sekce, na kterou bylo kliknuto. Vzhled připomíná harmoniku (*accordion*), proto toto jméno.

Autocomplete – Zajišťuje automatické vyplňování formulářových elementů. V praxi to vypadá tak, že se dají napsat seznamy nejvíce používaných slov (případně k nim přidávat ty hodnoty, co již byly vyplněny) a tyto hodnoty zobrazovat jako našeptávač.

Button – Element tlačítka. Pomocí tohoto elementu lze z inputů formuláře, které jsou typu radio nebo checkbox, udělat tlačítka a případně jej nastýlovat a přidat mu určitou funkčnost.

Datepicker – Element pro výběr data. Jedná se o klasický kalendář. Po rozbalení kalendáře je vidět aktuální měsíc (pokud není nastaveno jinak) a dny na výběr. Je možné listovat kalendářem jak do minulých měsíců (ale i let), tak do budoucích. Vybrané datum se vloží do vstupního pole formuláře podle způsobu formátování.

Dialog – Element dialog je obdobou javascriptového alertu. Jedná se dialogové okno, které zobrazuje informaci pro uživatele, např. že není přihlášen a vstupuje do sekce, kam mohou jen registrovaní uživatelé.

Progressbar – Element zobrazující procentuální průběh probíhané akce. Lze použít pouze v situacích, kdy systém může přesně vypočítat dobu.

Slider – Element posuvník. Tímto elementem lze nastavit hodnotu příp. nastavit rozmezí hodnot, ze kterých se má výsledná hodnota vytvořit. Například vygenerování náhodného čísla z rozmezí, které udávají 2 posuvníky. Je možno nastavit minimální, maximální hodnotu a kroky posunu.

Tabs – Element, který vytvoří z navigace takovou navigaci, která je tvořená ze záložek (*tabs*).

Kategorie efekty (*Effects*) obsahuje tyto efekty:

Effect – Jedná se o kolekci mnoha efektů, jako je např. třesení (*shake*), zneviditelnění (*blind*), pružení (*bounce*) a mnoha dalších. Efekt se aplikuje na objekt, který poté vykonává požadovanou vlastnost.

Show – Jedná se o metodu, která zviditelní objekt, který byl skrytý. Metoda přijímá parametr efekt, dobu trvání efektu a callback funkci.

Hide – Je opakem metody Show. Metoda skrývá objekt. Přijímá parametr efekt, dobu trvání efektu a callback funkci.

Toggle – Metoda, kterou tvoří kombinace metod Show a Hide. Tato metoda přepíná mezi stavy, tzn. pokud byl element skrytý, tak ho zobrazí a v opačném případě, kdy byl element zobrazen, ho skryje.

Color animation – Je efekt, který umožňuje animovat barevné přechody. Element s původní barvou se při vykonávání efektu obarví na barvu jinou.

Add class – Tento efekt umožňuje přidat elementu třídu. Je to obdoba nativního CSS (*Cascading Style Sheets*). Dovoluje změnit původně nadefinované vlastnosti a zaměnit je za nově nadefinované.

Remove class – Opačný efekt Add class. Je to obdoba nativního CSS (*Cascading Style Sheets*).

Toggle class – Efekt, který střídavě přidává a odebírá třídu elementu. Jedná se kombinaci Add class a remove class. Je to obdoba nativního CSS (*Cascading Style Sheets*).

Switch class – Efekt umožňující animovat přechod při přidávání či naopak odebírání třídy elementu. Je to obdoba nativního CSS (*Cascading Style Sheets*).

V kategorii pomocných nástrojů (Utilities) jsou nástroje na pozicování elementů.

Existuje několik verzí knihovny. Od jQuery UI 1.5 až po poslední stabilní verzi jQuery UI 1.9.1 (aktuální v době psaní této bakalářské práce). Jsou volně ke stažení z oficiálních stránek www.jqueryui.com.

Pomocí technologie jQuery UI bylo možné využívat hlavně zobrazovacích efektů *hide*, *show* a *toggle*. Dále je pomocí jQuery UI přidávána/odebírána třída u tabulek, která zajišťuje zvýraznění řádku, přes který se přejelo myší, resp. na kterém je aktuálně myš.



Obr. 4: Logo jQuery UI

1.6 jQPlot

jQPlot je javascriptový plugin, pomocí kterého se dají vykreslovat data do grafů. Existuje velké množství grafů, které lze vytvořit. Mezi ně patří sloupcové (*Bar charts*), pruhové (*Bar chars*), spojnicové (*Line charts*), výsečové (*Pie charts*), prstencové (*Donut charts*), bublinové (*Bubble charts*), burzovní (*Candlestick and Open Hi Low*

Close charts). Data se dají přijímat pomocí technologie Ajax, kdy se na server odešle požadavek (to vše na pozadí), skript požadavek zpracuje a zpět pošle výsledná, požadovaná data.

jQPlot je projekt s otevřeným zdrojovým kódem, dostupný pod dvěma licencemi. Ty jsou MIT (*licence se vznikem na Massachusetts Institute of Technology*) a GPL (*General Public Licence*). [9]

Technologie jQPlot byla použita pro vykreslování statistik. Ukázka statistiky je na obrázku 20.



Obr. 5: Logo jQPlot

2. Jazyk PHP

2.1 Historie jazyka

PHP z anglického PHP: Hypertext Preprocessor. Jeho historie se datuje k roku 1994, kdy byl poprvé vydán. Původně bylo PHP označení pro Personal Home Page (osobní domácí stránky). V roce 1994 byla napsána binární část Common Gateway Interface (CGI) v jazyce C. Tu napsal programátor Rasmus Lerdorf. Kód byl vytvořen k údržbě osobních domovských stránek. Později byl kód spojen s Form Interpreterem a právě toto spojení již umožňovalo komunikaci s databázemi. Tak začaly vznikat první dynamické webové stránky a aplikace. V roce 1995 byla vydána verze PHP 2, která měla základní funkčnost jako dnes známé PHP. Verze byla vydána veřejně, aby bylo možno kód zdokonalovat a najít co nejvíce chyb za pomoci veřejnosti. Vývojáři Zeev Suraski a Andi Gutsman v roce 1997 přepsali parser a položili tak základ PHP 3. Zároveň změnili původní označení pro PHP (Personal Home Page) na PHP: Hypertext Preprocessor. Verze vyšla roku 1998. Po vydání začali opět přepisovat jádro a společně vydali Zend Engine. Na tomto enginu je postavena verze PHP 4, která byla vydána roku 2000. Verze s označením PHP 5 byla vydána v roce 2004 a je postavená již na Zend Engine II. Hlavní revolucí PHP 5 bylo vylepšení možností objektově-orientovaného

způsobu programování. PHP 4 se nadále nevyvíjí a jedinou stabilní verzí se stává PHP 5. Tak tomu je od roku 2008. Od počátku PHP se tedy vystřídaly verze PHP 2, PHP 3, PHP 4, PHP 5, PHP 5.3, do které bylo přidáno static binding. V době vzniku této bakalářské práce existuje již PHP ve verzi 6, ale pouze jako betaverze. [2] [3]

2.2 Vlastnosti jazyka

Jedná se o jazyk serverový, skriptovací, interpretovaný (nikoli kompilovaný). Zpracování neprobíhá jako u JavaScriptu na straně uživatele (tedy v prohlížeči), ale na straně serveru. Jazyk PHP podporuje objektově-orientovaný způsob programování, ten se bohužel moc neujal, přestože má své příznivce. Slouží k programování dynamických webových stránek a aplikací. Mezi hlavní výhody patří jednoduchost jazyka, podobnost syntaxe s jazykem C. Jedná se o otevřený projekt s rozsáhlou podporou komunity. Snadno komunikuje s databázemi jako je MySQL, PostgreSQL, Oracle, MS SQL a dalšími. Je multiplatformní a lze jej tedy provozovat na většině dnes známých operačních systémů. Mezi nevýhody lze zařadit fakt, že podpora objektově-orientovaného způsobu programování není na tak vysoké úrovni, jako např. u jazyků Java, C# atd. PHP je aktivně vyvíjeno, ale na zpětnou kompatibilitu se již moc nehledí. Pro funkčnost PHP skriptů je vyžadován webový server. [2] [3]

Jazyk PHP byl použit k naprogramování stránek v kombinaci s jazykem HTML, komunikaci se serverem a databázovým systémem.



Obr. 6: Logo PHP

3. MySQL

3.1 Historie MySQL

MySQL je databázový systém vytvořený firmou MySQL AB, momentálně vlastněný společností Sun Microsystems, dceřinnou společností Oracle Corporation. Autory MySQL jsou Michael Widenius a David Axmark. První vydání bylo v roce 1995. [10]

3.2 Vlastnosti MySQL

Databáze MySQL, resp. program, kterému se říká DBMS (*DataBase Management System*), česky SŘBD (*Systém Řízení Báze Dat*) staví na tzv. relačním modelu dat. V tomto modelu jsou data uspořádána do tabulek a mezi těmito tabulkami jsou vztahy – relace. SŘBD umí základní práci s daty, tedy správa, čtení, změna dat, hledání apod. To vše probíhá na straně serveru. Pro komunikaci klienta s databázovým serverem existuje jazyk zvaný SQL (*Structured Query Language*). [3]

MySQL databázový systém (SŘBD společně s bází dat) spadá pod GPL (*General Public Licence*) licenci, ale je k dostání i s komerční placenou licencí. MySQL je multiplatformní a lze jej tedy snadno implementovat na Linux, Windows, ale i na další operační systémy. V současnosti je oblíbená technologie LAMP, což je zkratka Linux, Apache, MySQL, PHP, kde Linux je operační systém, na kterém běží webový server Apache a databázový systém MySQL a pomocí jazyku PHP se plní požadovaná funkčnost a probíhá komunikace (popř. PHP/SQL).

Technologie MySQL byla použita jako jazyk, který komunikuje s databázovým systémem. Pomocí MySQL jsou získávána, vkládána, upravována a mazána data, která jsou uložena v databázi aplikace.

3.3 Historie SQL

Historie SQL je datována roku 1974, kdy byl jeho prototyp prvně implementován společností IBM. V roce 1986 byl standardizován a stal se univerzálním nástrojem pro programování databází. [3]



Obr. 7: Logo MySQL

4. Jazyk HTML

4.1 Historie jazyka

HTML (*HyperText Markup Language*) prošel několika verzemi. První verze se objevila v roce 1990 a nesla pojmenování HTML 1.0. V roce 1993 byla představena neoficiální verze HTML+. Tato verze zahrnovala formuláře, tabulky a obrázky s popisem, ale neobsahovala formátování odstavců ani úpravu textů. Bylo v ní 78 elementů, kdy mnohé z nich už v dnešním HTML nenajdeme. Verze HTML 2.0 byla uvedena v roce 1994 a stala se oficiálním standardem. Obsahovala 49 elementů. V březnu roku 1995 se objevila verze HTML 3.0 a poprvé se zde vyskytly tabulky, možnost obtékání obrázků textem a matematické elementy. Necelý rok poté (v roce 1996) byla uvedena verze HTML 3.2, které obchází verzi HTML 3.0 a je považována za pravého nástupce HTML 2.0. Bylo přidáno 19 nových elementů a zůstaly tabulky a atributy obtékání textu z HTML 3.0. V roce 1997 byla uveřejněna verze HTML 4.0, které se během vývoje přezdívalo *Cougar*. Do této verze byla přidána podpora prvku *OBJECT*, který má velký význam pro vkládání obrázků a multimédií. Dále podporuje kaskádové styly CSS (*Cascading Style Sheets*), úpravy formulářů a tabulek, skriptování na straně klienta a internacionalizaci (podpora čtení jazyků zprava doleva atd.). [5]

4.2 Vlastnosti jazyka

HTML (*HyperText Markup Language*) je značkovací jazyk, který je definován v rámci SGML (*Standard Generalized Markup Language*), což je prapředek jazyka HTML. Slouží k vytváření dokumentů, ve kterých mohou být obsaženy hypertextové odkazy a pokročilejší formátování. Poskytuje prvky, s jejichž pomocí lze vytvářet, formátovat a upravovat webové stránky. K tomu slouží tyto způsoby:

- změna velikosti fontu, změna na tučné písmo, kurzívu
- vkládat obrázky, používat obrázkové mapy, vkládat animace typu GIF
- vytvářet formuláře
- vytvářet tabulky
- definovat barvu pozadí HTML dokumentu, tabulky, řádků tabulky nebo buněk tabulky
- vkládat odkazy na sekce v dokumentu, na dokumenty na jiných stránkách a na audio a video soubory. [5]

Jazyk HTML byl použit pro naprogramování webových stránek. Bylo toho docíleno v kombinaci s jazykem PHP a grafické stylování pomocí kaskádových stylů.

5. CSS

5.1 Historie CSS

CSS (*Cascading Style Sheets*) neboli tabulky kaskádových stylů vznikly za účelem formátování HTML dokumentů. V minulosti bylo formátování docíleno tak, že do HTML se přidávaly značky jako např. FONT, které se staraly o formátování. To ovšem ovlivňovalo vyhledávače a automatické zpracování stránek. HTML jazyk byl navržen tak, aby pouze formuloval strukturu dokumentu, nikoli jeho vzhled. K formátování nyní slouží kaskádové styly a většina značek, sloužících k formátování již z HTML vymizela. Jazyk vytvořila organizace W3C (*World Wide Web Consortium*) a doposud byly vydány 2 resp. 3 verze CSS, a to CSS1 a CSS2 a v roce 1997 byla dokončena verze CSS 2.1. Zároveň se aktuálně pracuje na verzi CSS3. [6] [11]

5.2 Vlastnosti CSS

Kaskádové styly složí k formátování HTML dokumentů. Je snadné rychle vyměnit vzhled dokumentu pouhou výměnou souboru `.css`. Je oddělena struktura HTML souboru a CSS nijak nezasahuje do jeho obsahu. Pouze se mění vzhled dokumentu. Kaskádové styly jsou sestavení pravidel, kdy každé pravidlo obsahuje selektor, což je identifikátor prvku a blok deklarací. V bloku deklarací jsou jednotlivé deklarace od sebe odděleny středníkem. Každá deklarace se skládá ze jména vlastnosti a hodnoty vlastnosti, oddělené od sebe dvojtečkou. Kaskádové styly nabízejí větší možnost formátování, než nabízelo HTML. Je snazší orientovat se v souboru, který slouží pouze pro formátování, než procházet celý kód, kde jsou jak HTML prvky, tak prvky formátování. [6] [11]

Technologie kaskádových stylů byla použita ke grafické podobě webových stránek.

Praktická část

6. Aplikace TaskManager

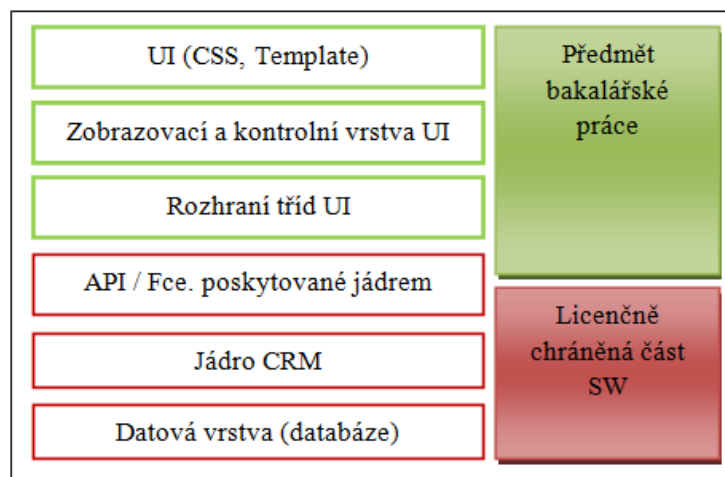
6.1 Seznámení s aplikací a její historie

Tato část bakalářské práce slouží jako uživatelská příručka a manuál k aplikaci. Aplikace TaskManager nevznikla pouze za účelem vytvořit a napsat tuto bakalářskou práci, ale také vytvořit produkt, který by dokázal obstát na trhu. Aplikace je tvořena primárně webovým rozhraním, kde je plná funkcionalita. Plnou funkcionalitou je rozuměna funkcionalita v rozsahu bakalářské práce. Produkt se bude nadále aktivně vyvíjet a rozšiřovat. Zároveň bude součástí několika dalších systémů, které budou tvořit jeden velký celek. K webovému rozhraní měla být k dispozici desktopová aplikace, která by měla minimální funkcionalitu. Sloužila by primárně pro ty uživatele, kteří jsou na cestách a nemohou být stále připojeni k internetu, ale potřebují pracovat na projektech. Po mnoha konzultacích s firmou se od desktopové aplikace ustoupilo s tím, že aplikace by neměla využití, protože všichni zaměstnanci firmy mají notebooky s přenosným internetovým připojením. V příloze je slovy konzultanta popsána historie z jeho pohledu.

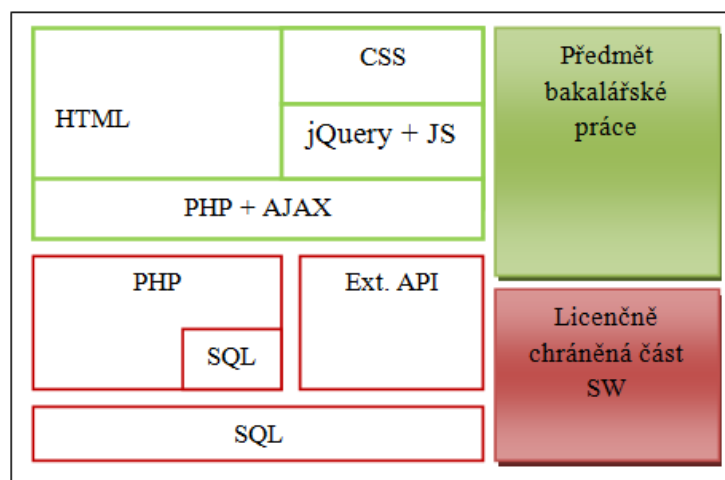
Aplikace TaskManager je určena pro uživatele, kteří by rádi měli pod kontrolou vlastní čas, který tráví nad svými projekty, úkoly a pracemi s nimi souvisejícími. Primární funkcionalitou je tedy měřit čas práce. Pro registraci uživatele je nutno zaslat e-mail s kontaktními údaji a přihlašovacím jménem. Příslušný pracovník údaje zpracuje do databáze a po úspěšném zpracování je systémem odeslán vygenerovaný e-mail s přihlašovacími údaji zákazníka. Email obsahuje pouze přihlašovací uživatelské heslo, které bylo vygenerováno systémem. Heslo jde kdykoliv změnit v sekci profilu uživatele.

Aktuální verze softwaru je navržena na firemní potřeby. Z toho důvodu není momentálně možné zakládat pouze osoby, ale každou osobu je nutno vložit do firmy. Do budoucna se počítá s rozšířením této vlastnosti na možnost registrovat uživatele bez nutnosti být ve firmě. V rámci firmy existuje několik stupňů rolí uživatelů. Momentálně to jsou 3 stupně a to role vedoucího, role projektového manažera a role zaměstnance. Role se budou vyvíjet a rozšiřovat podle potřeb zákazníků.

Na obrázcích 8 a 9 je zobrazen diagram vrstev a technologické rozložení aplikace. Z obrázku je patrné, které části byly poskytnuty od dodavatele, a které byly vytvářeny v rámci bakalářské práce.



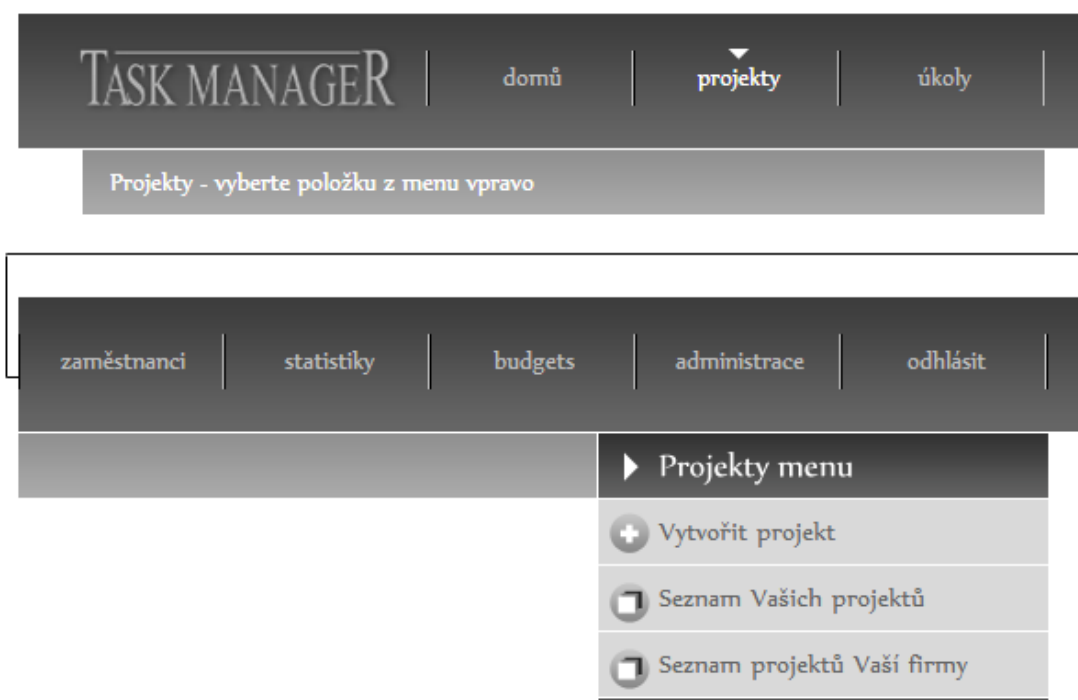
Obr. 8: Diagram vrstev aplikace



Obr. 9: Diagram technologického rozložení aplikace

6.2 První příchod na webové stránky

Při prvním příchodu na webové stránky www.vspszone.cz uživatele vítá příjemné prostředí s velmi omezenou funkcionalitou. Uživatel se z domovské stránky dozví pouze obecné informace o aplikaci. Pokud vlastní přihlašovací údaje, stačí se přihlásit v sekci „přihlášení“ a vyplnit je. Kontaktní údaje jsou uživateli zasílány e-mailem po vložení do databáze. Po přihlášení se zpřístupní sekce, do kterých má uživatel, dle své přidělené role, přístup. Stránky jsou rozčleněny na horní navigaci, na obsah a menu, která jsou ve většině sekcí (viz obr. 10).



Obr. 10: Rozložení navigace a aplikace

6.3 Horní navigace – hlavní menu

V hlavním menu je osm sekcí resp. odkazů, a jsou to tyto:

- domů
- projekty
- úkoly
- zaměstnanci
- statistiky
- budgets
- administrace
- přihlásit / odhlásit

(viz obr. 10), kde sekce projekty, úkoly, zaměstnanci a administrace mají své vlastní podmenu.

6.3.1 Sekce „domů“

Položka menu je zpřístupněna vždy a všem uživatelům, ať nepřihlášeným či přihlášeným, a to se všemi úrovněmi práv. Pokud uživatel nemá dostatečná práva a pokusí se dostat do sekce s vyššími právy, zobrazí se upozornění, že do této sekce nemá přístup a dostatečná práva. Znepřístupnění určitých položek menu bylo docíleno přidáním třídy *disable* k elementu *odkaz* u *nesetříděného seznamu* a pomocí javascriptového kódu, se danému elementu odebrala vlastnost *href*, která odkazuje na cílový soubor. Zároveň bylo v kaskádových stylech docíleno zobrazení varovného okna po přjetí myši přes znepřístupněný element menu (viz obr. 11).



Obr. 11: Nedostačující oprávnění

Sekce/stránka domů je titulní stranou aplikace, kde jsou informace o aplikaci a postup pro registraci.

6.3.2 Sekce „projekty“

Sekce projekty je určena pro správu projektů. Obsahuje menu, ve kterém jsou základní možnosti, pro práci na projektech. Do sekce mají přístup všichni registrovaní uživatelé s tím rozdílem, jaké možnosti menu mají povoleny. Jsou zde 3 základní položky menu a to *vytvořit projekt*, *seznam Vašich projektů* a *seznam projektů Vaší firmy*, ve které pracujete. Projekty mohou vytvářet pouze uživatelé s právy typu *vedoucí* a *projektový manažer*. Posledním typem uživatele je uživatel *zaměstnanec*, který má nejnižší práva. Prohlížet projekty mohou všechny tři typy uživatelů.

V položce *vytvořit projekt* je formulář pro nový projekt. Je nutno zadat název projektu, vyplnit krátký popis projektu, zapsat cenu v Kč (pouze informativní položka v rámci firmy), určit, zda projekt po vytvoření bude veřejný či privátní, zadat status projektu a vybrat zaměstnance, kterému projekt chcete přiřadit. Pokud nastavíte projektu status *neveřejný*, potom projekt zaměstnanec neuvidí do té doby, než změníte stav na *veřejný*. Kontrola zadaných hodnot je prováděna pomocí javascriptového

validátoru, vytvořeného speciálně pro tento typ formulářů na stránce. Kontroluje se, zda byl zadán název projektu, jeho popis, cena. Ostatní hodnoty mají již výchozí hodnotu, proto je netřeba kontrolovat. Ke vstupním polím se zobrazuje malá nápověda. Formulář pro vytvoření projektu je na obrázku 12.

Info o projektu Zadejte název projektu

Název projektu:

Popis projektu:

Cena projektu:

Veřejný: ☒ ANO

Status projektu: open

Přiřadit zaměstnanci: Jan Wanke

Odeslat

Obr. 12: Formulář přidání projektu

V příloze č. 3 je k vidění ukázka práce ve třídě, která se stará o vykreslení formulářových prvků a celých formulářů. Konkrétně se jedná o formulář přidání projektu. Funkci se nejprve předá identifikace uživatele, který je aktuálně přihlášen. Identifikace uživatele se předává pomocí parametru. Parametr je dále zpracováván pro zjištění adresy firmy, ve které uživatel pracuje. Po zjištění adresy, kde uživatel pracuje, se načítají pomocí dalšího volání funkce všichni zaměstnanci, kteří pracují v konkrétní firmě. Do pole unikátních identifikátorů se vkládají čísla zaměstnanců tak, aby se neduplikovali v poli. Vytvoří se pole všech identifikačních čísel zaměstnanců

pracujících ve firmě. Začne se vykreslovat formulář, prvek po prvku. Následuje procházení všech zaměstnanců v systému, kteří se načetli do dalšího pole. Jedná se o pole objektů. Při procházení se porovnávají identifikace uživatelů s identifikacemi, které jsou uloženy v poli, ve kterém jsou jen čísla zaměstnanců. Při shodě se do výběrového pole (pomocí kterého se přiřazuje projekt uživatele) vloží identifikátor a jméno s příjmením zaměstnance. Totožná práce probíhala na jednotlivých částech UI, kde docházelo k postupnému napojení rozhraní jádra na layout obohacený javascript, jQuery, AJAX a HTML.

Položka *seznam Vašich projektů* obsahuje projekty, které Vám byly přiřazeny a mají status *veřejný*, projekty jsou *ověřeny* a *schváleny* (viz dále). Je zde vidět jméno projektu, jeho popis a status. Poslední sloupec slouží k dokončení projektu. Jakmile bude projekt hotov, stisknete tlačítko dokončit a status projektu se změní na *dokončený*. Poté tlačítko zmizí a na projektu není možno dále pracovat.

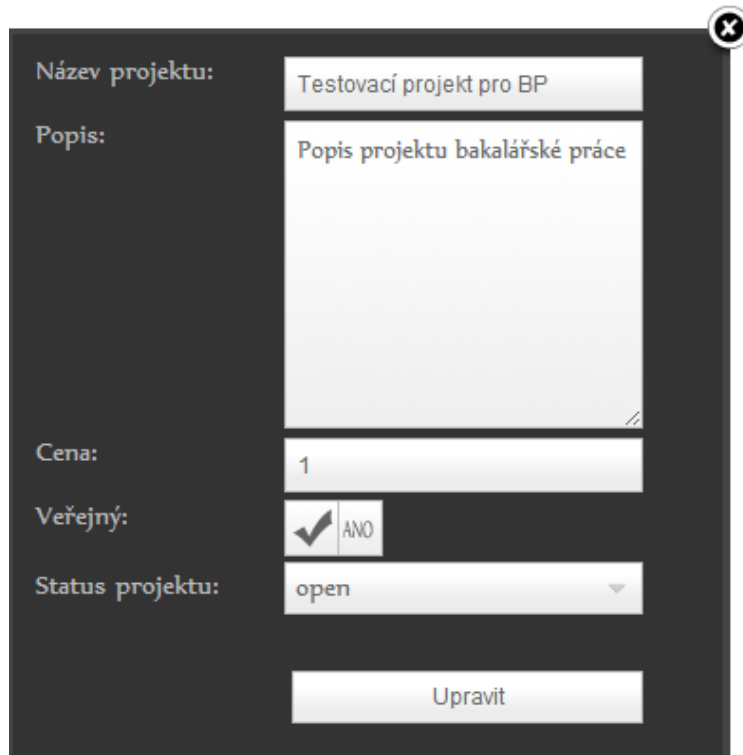
Poslední položkou menu je *seznam projektů Vaší firmy*. Zde jsou vidět veškeré projekty, které jsou ve firmě vytvořeny. U každé tabulky, která se na stránkách vyskytuje, lze řadit data vzestupně/sestupně díky jQuery pluginu *sortable*.

Název projektu	Popis projektu	Schválený	Ověřený	Věřejný	Status	Operace
Testovací projekt 1	popis testovacího projektu 1	1	0	1	open	     
Testovací projekt 2	popis testovacího projektu 2	0	1	0	run	     
Testovací projekt 3	popis testovacího projektu 3	0	0	0	to finish	     


Obr. 13: Seznam projektu Vaší firmy

Kromě podobné tabulky (viz obr. 13), jako je u seznamu Vašich projektů, je zde navíc sloupec *schválený*, *ověřený* a *veřejný*. Každý zobrazuje stav projektu. Aby bylo možno na projektu pracovat, musí se schválit, ověřit a změnit status na veřejný, pokud

tak nebylo učiněno při vytváření projektu. Změna probíhá kliknutím na tlačítko ✖ v případě schvalování projektu (tato ikona je viditelná, pokud je projekt neschválen). Po schválení se ikona změní na ikonu ✔ a stav projektu je nyní schválený. Výchozí ikona u stavu ověřený je 🧐 a kliknutím na ikonu proběhne ověření projektu a ikona se změní na 👍. V tuto chvíli je stav projektu schválený a ověřený. Pokud z jakýchkoliv příčin dojde ke změně stavu, např. bylo zjištěno neplatné ověření, je možno stavy dalším kliknutím vrátit do výchozího stavu. Stav je viditelný z aktuální ikony. Dalšími možnostmi u projektů je upravování a mazání projektu. Úprava probíhá po stisku tlačítka ✎. Objeví se dialogové okno s možností úprav názvu projektu, popisu, ceny, statutu veřejný/neveřejný a statusu, v jaké fázi se projekt momentálně nachází (viz obr. 14). Dialogové okno je zobrazováno pomocí technologie simplemodal, která není v teoretické části vysvětlena, ani nijak popsána. Jedná se o technologii, která využívá knihoven jQuery a vlastních javascriptových knihoven k vykreslení dialogového okna, do kterých se pomocí jazyka HTML vloží požadované elementy. Na obrázku 14 je vidět dialogové okno, ve kterém je formulář s hodnotami upravovaného projektu.





Obr. 14: Dialogové okno - úprava projektu

Po změně požadovaných údajů, stisknutí tlačítka upravit a úspěšné kontrole dat systémem se data zaktualizují a budete přesměrováni zpět na seznam projektů Vaší firmy, kde bude vidět Vámi provedená změna. Kontrola dat probíhá opět za pomoci skriptu k validování formulářových polí. Validace probíhá u všech formulářů obdobně. Posledním tlačítkem je tlačítko dokončit s ikonou . Toto tlačítko slouží k dokončení projektu v případě, že uživatel, kterému byl projekt přiřazen, není schopen projekt dokončit sám. Všechny výše jmenované funkce může provádět pouze uživatel s právy *vedoucí* a v případě úprav a mazání i *projektový manažer*. Uživatel *zaměstnanec* může pouze nahlédnout. Tabulka je doplněna o nápovědu, která se nachází pod vlastním menu sekce projektů a obsahuje vysvětlení funkcionality tlačítek.

6.3.3 Sekce „úkoly“

Sekce úkoly je určena ke správě úkolů. Obdobně jako u sekce projekty je i zde možnost vytvoření úkolu na projektu, a to se stejnojmenným názvem *vytvořit úkol*.

Vytvořit úkol je první položka menu. V té je opět krátký formulář s vyžadovanými daty, jako je název úkolu, popis úkolu a status. Dále je zde navíc pole s výběrem projektu, ke kterému se má přiřadit. Po odeslání a systémové kontrole dat se je vytvořen. Máte možnost vytvořit další nebo se přesunout do *seznamu úkolů na Vašich projektech*. Vytvářet mohou opět výše privilegovaní uživatelé *vedoucí* a *projektový manažer*.

Druhou položkou menu je *seznam úkolů na Vašich projektech*. Zde jsou evidovány Vaše projekty a k nim příslušné úkoly. Vše je rozděleno do tabulky. Ke každému úkolu je evidován celkový čas práce. Ten je výsledkem všech Vašich prací na úkolu. Výše privilegovaní uživatelé mohou úkoly upravovat, případně mazat. K tomu jsou určena stejná tlačítka jako u projektů  a . Novým tlačítkem je **S**, což je seznam prací, které proběhly. Na práci se uživatel přesune stiskem tlačítka:



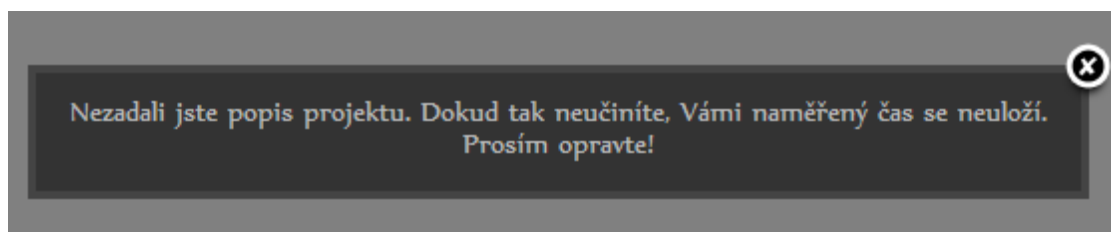
Obr. 15: Začít pracovat na úkolu

Následně je uživatel přesměrován již ke konkrétní práci (viz obrázek 16).



Obr. 16: Měření času na úkolu

Pro spuštění měření je nutno stisknout tlačítko start u stopek. Horní – aktuální čas, je čas, který uživatel strávil/tráví momentálně na práci. Dolní – celkový čas, je čas, který uživatele informuje, kolik ho strávil na celém úkolu. Stejný čas je viditelný i v tabulce úkolů. Konec měření nastává stiskem tlačítka stop. Tlačítko stop slouží i jako pozastavení práce a stopky lze opět spustit tlačítkem start. Pro ukončení práce a uložení do systému je nutno stopky zastavit, vyplnit popis práce a poté dát uložit. V případě, že nevyplníte popis práce, budete vyzváni, ať tak učiníte pomocí dialogového okna (viz obr. 17). Tlačítko stop je ve výchozí poloze deaktivováno a lze jej stisknout až po stisknutí start. Stejná situace je i u tlačítka uložit, které bez prvotního spuštění nelze stlačit a odeslat. Po spuštění start se zpřístupní stop. Po ukončení měření se aktivuje možnost uložení (v případě, že uživatel opět nespustí měření – to se situace opakuje). Stopky jsou ovládány skriptem v jazyku javascript, který se stará o vykreslování hodnot do vstupních polí formulářového prvku typu *input*.




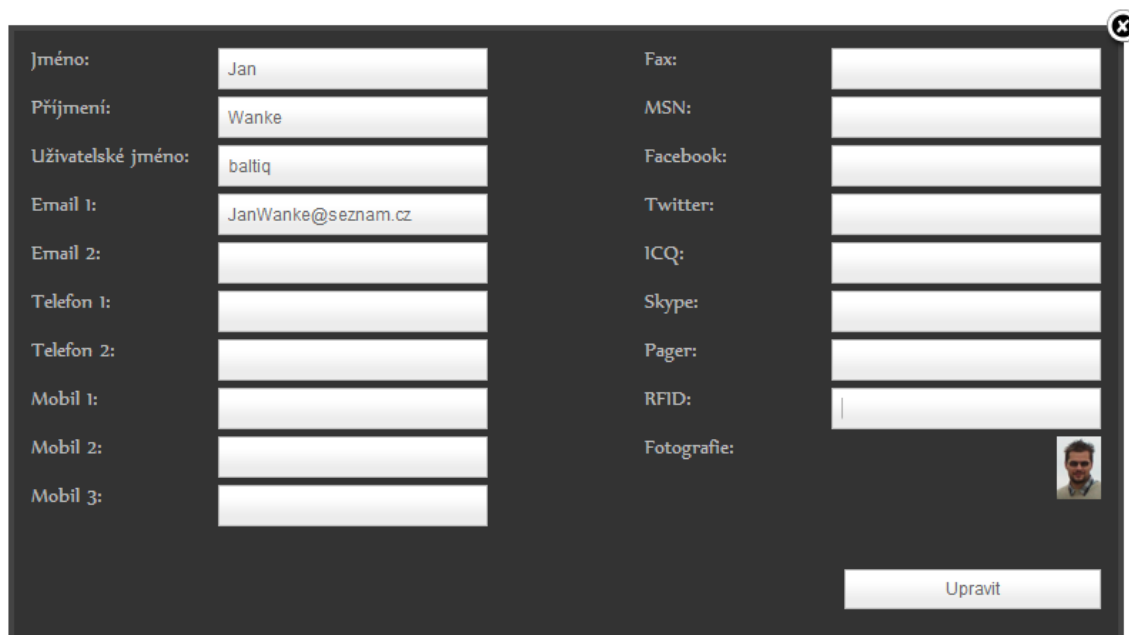
Obr. 17: Dialogové okno - nevyplnění popisu práce

Po úspěšném uložení do systému budete přesměrováni na seznam úkolů Vašich projektů, kde se můžete podívat na seznam prací a k nim jejich dobu práce pod tlačítkem **S**.


6.3.4 Sekce „zaměstnanci“

V sekci jsou dvě položky menu, a to *seznam Vašich zaměstnanců* a *správa rolí Vašich zaměstnanců*.

V první položce *seznam Vašich zaměstnanců* jsou v tabulce zaměstnanci, kteří pracují u Vás ve firmě. Je zde zobrazeno jméno a příjmení zaměstnance, jeho tituly, uživatelské jméno a e-mail. V posledním sloupci jsou operace s kontaktem. Tlačítko profilu **P** umožňuje zobrazit profil uživatele a jeho kontaktní údaje nebo přidat nové pomocí tlačítka . Jedná se o přidání mobilních telefonů, pevných linek, skype, facebook, twitter účtů, apod. (viz obr. 18).



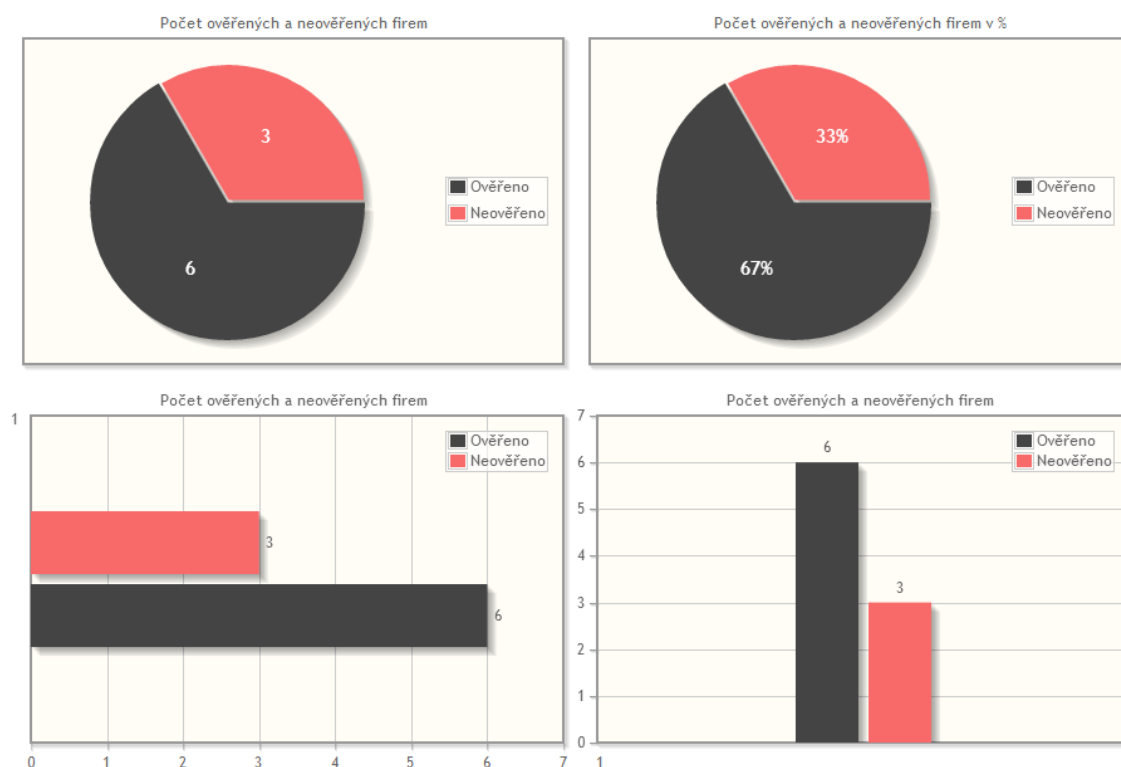
Obr. 18: Dialogové okno - úprava profilu

Položka *správa rolí Vašich zaměstnanců* slouží k úpravě práv zaměstnanců v rámci firmy, kterou mohou provést pouze uživatelé s rolí *vedoucího*. Je zamezeno, aby si uživatel přiřadil vyšší práva, než jaká doposud má. Tím se zamezí uživateli v přidělení si vyšších práv, než která aktuálně má. Změna práv probíhá po stisku tlačítka úprav  u uživatele, kterému chcete práva pozměnit. Otevře se dialogové okno

s výběrem rolí a jménem zaměstnance. Po vybrání a potvrzení role se uživateli změní jeho práva a vy budete přesměrováni zpět na tabulku s možností změn práv uživatelů.

6.3.5 Sekce „statistiky“

Sekce je určena pro vykreslování statistik. Stačí pouze vybrat ze seznamu předem definovaných statistik. Zpět se vrátí vykreslený graf popř. statistika s krátkým popiskem. Vykreslování statistik je zajištěno technologií jqPlot. Po vybrání statistiky z možností výběrového okna, se na server odešle AJAX požadavek na vrácení hodnot ze systému. Databázový systém zpracuje požadavek a odešle požadovaná data. Data se předávají do javascriptového kódu, kde se jednoduchým nastavením určí, o jaký graf se jedná a jaké je jeho grafické formátování. Skript se ukončuje a statistika se vykresluje do grafu. Jsou k vidění 2 typy grafů. Dva koláčové grafy a dva sloupcové. První koláčový zobrazuje počet firem v nominální hodnotě. Druhý zobrazuje procentuální množství. Třetí a čtvrtý graf zobrazuje stejná data jako první s rozdílem, že u třetího jsou sloupce orientovány horizontálně a u čtvrtého jsou vertikálně (viz obr. 19).



Obr. 19: Statistiky - počet ověřených firem

6.3.6 Sekce „budget“



Sekce je nad rámec této bakalářské práce, a proto zde nebude popsána. S touto sekci se počítá do budoucna a bude obstarávat funkcionalitu ohledně financí na projektech apod.

6.3.7 Sekce „administrace“



Do sekce je striktně omezen přístup a slouží pouze správcům webové aplikace. Sekce obsahuje hned několik položek menu, mezi které patří přidání nové firmy, přidání adresy již ke stávajícím firmám, v případě, že se firma rozrostla o další pobočku, vytvořit nového zaměstnance a toho přiřadit do firmy, nechat si vypsat seznam všech uživatelů evidovaných v databázi, nechat vypsat všechny projekty a úkoly uložené v systému, kdo je na kterém projektu a případná správa uživatelských rolí. Správci webové aplikace nebudou nijak zasahovat do vnitřní organizace projektů, jejich přiřazování apod. Výpisy jsou pouze kontrolní s možností zásahu v případě porušení zákonů či etických kodexů.

Položka *přidat firmu*, slouží k vytvoření nové firmy s názvem, adresou a identifikačními čísly. K vytvoření firmy slouží krátký formulář. Před odesláním se data kontrolují systémem a v případě, že je vše v pořádku, firma se vytvoří. Formulář je na obrázku 20 vlevo.

Položka *přidat adresu firmy* slouží k vytvoření nové adresy firmy. Jedná se o stejný formulář (na obrázku 20 vpravo) jako v případě přidání firmy s tím rozdílem, že na prvním místě se vybere, ke které firmě se přidává adresa a již se nezadává jméno společnosti. Na obrázku 20 je vidět rozdíl mezi formulářem, který přidá novou firmu a formulářem, který přidá již k vytvořené firmě pouze adresu a její identifikaci.

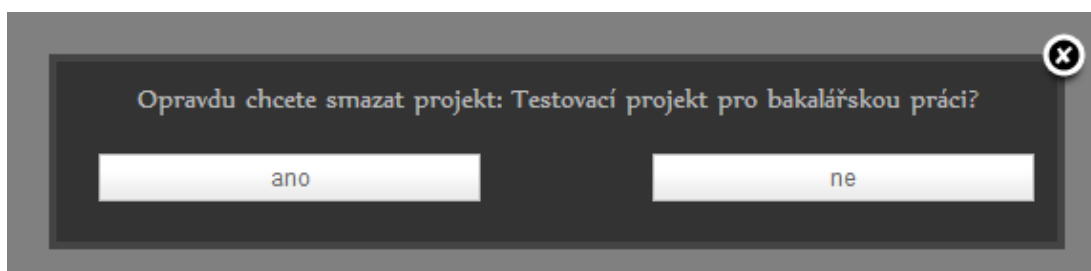
	Info o firmě	Info o firmě
Název firmy:	<input type="text"/>	Vyberte firmu z možností ▾
Země:	 Česká republika ▾	 Česká republika ▾
Kraj:	Ústecký kraj ▾	Praha ▾
Okres:	Okres Ústí nad Labem ▾	Praha ▾
Město:	Ústí nad Labem ▾	Praha 1 ▾
Ulice:	<input type="text"/>	<input type="text"/>
Číslo ulice:	<input type="text"/>	<input type="text"/>
Číslo domu:	<input type="text"/>	<input type="text"/>
PSČ:	<input type="text"/>	<input type="text"/>
IČO:	<input type="text"/>	<input type="text"/>
DIČ:	<input type="text"/>	<input type="text"/>
Plátce DPH:	<input checked="" type="checkbox"/> ANO	<input type="checkbox"/> NE

Obr. 20: Rozdíl formulářů - přidat firmu (vlevo) / přidat adresu firmy (vpravo)

Položka *přidat zaměstnance* slouží k vytváření zaměstnanců. V aktuální verzi se o vytváření zaměstnanců starají správci webu. Do budoucna se počítá s rozšířením o roli sekretářka, která bude v každé firmě, pokud tak firma bude chtít, případně jedna sekretářka z řad správců, která se o tyto požadavky bude starat. Vytvoření zaměstnance probíhá pomocí formuláře, kam se zadávají uživatelské tituly, jméno, příjmení, uživatelské jméno a e-mail. Systém automaticky vygeneruje heslo a na zadaný e-mail, po vložení do databáze odešle informativní email s uživatelským jménem a vygenerovaným heslem. Heslo lze po přihlášení změnit v sekci profilu, případně v seznamu zaměstnanců, kde je tlačítko upravit  pouze u přihlášeného zaměstnance, v případě role zaměstnanec. Privilegovaní uživatelé mohou měnit ostatní uživatele, ale pouze v rámci firmy. Vytvořený uživatel je zobrazen v seznamu všech uživatelů v systému a v sekci operací se objeví ikona  pro přiřazení kontaktu do firmy. To probíhá pomocí dvou výběrových oken, kde v prvním je jméno firmy a v druhém je adresa, případně adresy, pokud má firma více poboček či adres.

Položka *seznam všech firem* vypisuje tabulku evidovaných firem v systému. Jejich názvy, všechny adresy, které jsou ke konkrétní firmě přiřazeny, identifikační čísla, statusy schválená/neschválená (schválená v případě, že je v systému nadefinováno schválení těchto nových záznamů nadřazeným), ověřená/neověřená (údaje jsou ověřeny z nezávislého zdroje/nezávislým uživatelem), od kdy je firma založena a možné operace s firmou. Mezi operace opět patří možnost firmu upravit tlačítkem ✎, smazat tlačítkem ✖, změnit její status z neověřené 🤝 na ověřenou 👍 a změnit status z neschválené ✖ na schválenou ✅ a naopak. K tabulce je uvedena legenda popisující funkce jednotlivých tlačítek, která je orientována pod menu, napravo od tabulky.

Položka *seznam všech zaměstnanců* vypisuje tabulku všech uživatelů, kteří jsou evidováni v systému. Jsou k vidění kontaktní údaje a operace obdobně jako u sekce uživatelů v sekci zaměstnanci. Lze zobrazit profil 👤, upravit uživatele ✎, případně ho smazat ✖. Mazání zaměstnance je ještě nutno potvrdit v dialogovém okně, zda opravdu souhlasíte s mazáním (viz obr. 21).






Obr. 21: Dialogové okno - potvrzení smazání projektu


K tabulce je nápověda znovu vysvětlující výše popsané funkce, která je orientována pod menu, napravo od tabulky.

Položka *seznam všech projektů* vypisuje tabulku projektů uložených v systému, jejich popisy, statusy schválený/neschválený, ověřený/neověřený a status projektu. Poslední sloupec slouží opět k práci s projekty. Projekt lze upravovat ✎, mazat ✖, změnit status schválení ✅ ✖, změnit status ověření 👍 🤝 a dokončit 🏁. Tento výpis slouží opět jako kontrolní výpis, do kterého správci zasáhnou pouze v případě porušení zákonů nebo etiky. Režie projektů je čistě interní záležitost firmy a nastavuje se pomocí rolí, případně dílčích politik, které jsou otázkou customizace aplikace. K tabulce je

uvedena legenda popisující funkce jednotlivých tlačítek, která je orientována pod menu, napravo od tabulky.

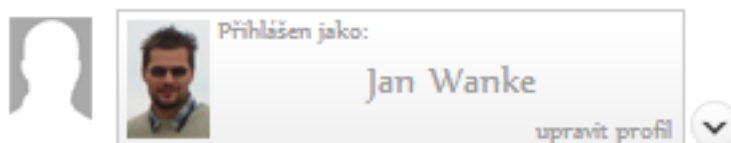
Položka *seznam všech úkolů*, vypisuje všechny úkoly, které jsou uloženy v systému. Je zobrazen název úkolu, jeho popis, kdy byl úkol vytvořen, kolik bylo na úkolu odpracováno času, jeho statusy a operace upravit , mazat  a přejít na seznam prací , které byly na úkolu provedeny. Zde je zobrazen výpis s popisy práce, které se zadávají při ukládání do systému a odpracovaný čas. K tabulce je nápověda znovu vysvětlující výše popsání funkce, která je orientována pod menu, napravo od tabulky.

Položka *projekty u všech zaměstnanců* zobrazuje výpis se všemi uživateli a projekty, které jim jsou přiřazeny a na kterých pracují.

Položka *správa rolí všech zaměstnanců* slouží ke správě a úpravě rolí zaměstnanců. Oproti správě rolí, která se nachází v sekci zaměstnanci, lze zde přiřadit i roli správce, která z bezpečnostních důvodů není povolena v rámci firem. Správa se provádí kliknutím na tlačítko spravovat role  po otevření dialogového okna vybrat/změnit roli u konkrétního uživatele a potvrdit.

6.3.8 Sekce „přihlásit / odhlásit“

Tato sekce slouží k přihlášení, případně k odhlášení uživatele. Je to první resp. druhá sekce, do které by se měl uživatel podívat po přečtení informací na úvodní stránce a po přijetí uživatelských údajů. Po uživateli je požadováno uživatelské jméno a heslo. Po úspěšném přihlášení se v pravém dolním rohu objeví profil uživatele s několika základními informacemi, do kterých patří jméno přihlášeného uživatele, tituly, pokud uživatel nějaké má a profilové fotografie uživatele. Pokud uživatel nenahraje vlastní fotografii, použije se výchozí systémový obrázek, který je možno následně nahradit vlastní fotografií. Poslední věcí v tomto okně profilu je odkaz k upravení profilu (viz obr. 22). Zde může uživatel upravit své osobní informace.



Obr. 22: Profil po přihlášení + výchozí profilová fotografie

7. Závěrem

7.1 Problémy při vypracovávání

V průběhu práce na aplikaci se vyskytlo několik úskalí a problémů. Mezi ty největší lze zařadit změnu datového modelu a následné úpravy API ze strany dodavatele. To výrazně zpomalilo práci, jelikož celé jádro aplikace a poskytované funkce jsou robustní a skýtají velkou škálu uživatelských nastavení a konfigurací pro následnou customizaci. Dalším problémem, který se dal dostatečně rychle překonat, bylo seznámení se s veškerými technologiemi, které jsou pro aplikaci klíčové. S většinou technologií jsem měl již zkušenosti, ale byly použity i ty, které jsem viděl prvně. Posledním problémem bylo odladění aplikace na webových prohlížečích, konkrétně na Internet Exploreru. Problém byl se zobrazování, některých grafických elementů knihovny jQuery UI a dialogových oken knihovny jQuery.

7.2 Bezproblémové části práce

Praktický žádný problém nebyl s grafickým návrhem aplikace, který jsem celý vypracoval samostatně. To pramenilo z bohatých zkušeností s grafickými aplikacemi, návrhy grafiky pro jiné webové stránky nebo jiné propagační materiály apod. Veškerá grafika je buď vytvořena speciálně pro tuto aplikaci (v případě webových stránek, stylizace formulářových prvků apod.), nebo upravená tak, aby ladila k vytvořené grafice (stylizace stopek apod.).

7.3 Přínos práce pro autora

Hlavní přínosem jsou nově nabyté zkušenosti při objektově-orientovaném programování v PHP. Tato práce zásadně změnila můj přístup k programování, způsobu myšlení a navrhování aplikací. Dosavadní způsob byl neefektivní, co se týče možnosti pozdějších úprav, možnosti zaimplementování nových částí aplikace a celková její údržba. Při práci jsem nabyt nové vědomosti, kterých lze do budoucna využít. Rozšířil jsem si obzory ohledně návrhů grafiky, návrhu a programování webových stránek, adresářové struktury, všech použitých technologií a jejich vzájemné spolupráci na stránkách. Poslední věcí, kterou mi tato práce dala, je schopnost pracovat s cizím API/funkcemi aplikace, dokázat řešit případné vzniklé problémy, nedostatky

v dokumentaci a komunikovat s dodavatelem, dále poskytnuté funkcionality rozvíjet a přidávat vlastní.

7.4 Stanovené cíle, splněné a nesplněné úkoly

Cílem této bakalářské práce bylo vytvořit produkt, který by obstál na trhu proti konkurenci. Zda se tak povedlo, ukáže teprve čas. Práce měla obohatit autora o nové zkušenosti s technologiemi a jazyky popsány v zadání a teoretické části. Autor obdržel datový model a jádro, na které začal nabalovat a programovat vlastní funkcionalitu odpovídající zadání práce. Bylo nutno navrhnout veškerou grafickou podobu grafického rozhraní, webových stránek, webových komponent, jako jsou tlačítka, nastýlované formulářové elementy, logo aplikace, rozvržení stránek apod. V průběhu programování došlo k několika změnám v oblasti grafiky, které znamenaly většinou posun kupředu za cenu mírné ztráty času. Výsledkem by mělo být přehledné a příjemné uživatelské prostředí pro snadnou práci na projektech. Dále bylo nutno vytvořit funkcionalitu, která propojí grafické rozhraní, jádro aplikace, datový model a vlastní vytvořenou funkcionalitu. Ze vzájemných konzultací s P. Mojžíšem bylo do jádra zavedeno několik dalších funkcí, se kterými jsem přišel. Dále bylo nutné vytvořit onu veškerou funkcionalitu, co se týče vytváření projektů, vytváření úkolů k nim, měření času stráveného na konkrétním úkolu, veškeré funkce úprav, mazání, změn statusů, jejich zviditelňování, validování veškerých formulářů, kontrola odesílaných dat z formulářů na straně serveru, pokud prošla validací apod. Práce měla naučit autora novým dovednostem a znalostem v oblasti programování webových stránek, grafiky a propojování technologií a jejich vzájemné komunikaci a spolupráci k vytvoření komplexního produktu.

Všechny výše zmíněné body byly splněny až na jeden. Tím je desktopová aplikace, u které (jak již bylo psáno) bylo po konzultacích stanoveno, že by pro firmu neměla přínos. To je způsobeno tím, že všichni pracovníci firmy vlastní notebook s přenosným internetem. Desktopová aplikace měla sloužit pouze uživatelům, kteří by byli mimo dosah internetu, což v tomto případě ve valné většině odpadá.

Seznam použité literatury a odkazů:

- [1] [online]. [cit. 2013-01-06]. Dostupné z: <http://api.jquery.com/>
- [2] PROCHÁZKA, David. *PHP 6: začínáme programovat*. 1. vyd. Praha: Grada, 2012, 183 s. Průvodce (Grada). ISBN 978-80-247-3899-4.
- [3] PONKRÁC, Miloslav. *PHP a MySQL: bez předchozích znalostí: [průvodce pro samouky]*. Vyd. 1. Brno: Computer Press, 2007, 221 s. Průvodce (Grada). ISBN 978-80-251-1758-3.
- [4] KOEGH, James. *Java bez předchozích znalostí: průvodce pro samouky*. Vyd. 1. Brno: Computer Press, 2005, 274 s. Průvodce (Grada). ISBN 80-251-0839-2.
- [5] PÍSEK, Slavoj. *HTML a XHTML: začínáme programovat: podrobný průvodce začínajícího uživatele*. 1. vyd. Praha: Grada, c2003, 255 s. Průvodce (Grada). ISBN 80-247-0571-0.
- [6] STANÍČEK, Petr. *CSS Kaskádové styly: kompletní průvodce*. 2. vyd. Brno: Computer Press, 2003, 178 s. Průvodce (Grada). ISBN 80-722-6872-4.
- [7] [online]. [cit. 2013-01-06]. Dostupné z: <http://api.jqueryui.com/>
- [8] [online]. [cit. 2013-01-06]. Dostupné z: <http://www.tinymce.com/wiki.php>
- [9] [online]. [cit. 2013-01-06]. Dostupné z: <http://www.jqplot.com/docs/files/usage-txt.html>
- [10] [online]. [cit. 2013-01-06]. Dostupné z: <http://dev.mysql.com/doc/>
- [11] [online]. [cit. 2013-01-06]. Dostupné z: <http://www.w3.org/Style/CSS/>
- [12] BAŠE, Ondřej. *jQuery pro neprogramátory: průvodce využitím knihovny jQuery UI*. 1. vyd. Brno: Computer Press, 2012, 368 s. ISBN 978-80-251-3750-5.
- [13] [online]. [cit. 2013-01-06]. Dostupné z: https://developer.mozilla.org/en-US/docs/AJAX/Getting_Started
- [14] [online]. [cit. 2013-01-06]. Dostupné z: http://www.oreillynet.com/pub/a/java-script/2001/04/06/js_history.html

Seznam obrázků:

Obr. 1: Logo JavaScript.....	10
Obr. 2: Logo AJAX.....	11
Obr. 3: Logo jQuery.....	12
Obr. 4: Logo jQuery UI.....	15
Obr. 5: Logo jqPlot.....	16
Obr. 6: Logo PHP.....	17
Obr. 7: Logo MySQL.....	19
Obr. 8: Diagram vrstev aplikace.....	24
Obr. 9: Diagram technologického rozložení aplikace.....	24
Obr. 10: Rozložení navigace a aplikace.....	25
Obr. 11: Nedostačující oprávnění.....	26
Obr. 12: Formulář přidání projektu.....	27
Obr. 13: Seznam projektu Vaší firmy.....	28
Obr. 14: Dialogové okno - úprava projektu.....	29
Obr. 15: Začít pracovat na úkolu.....	30
Obr. 16: Měření času na úkolu.....	31
Obr. 17: Dialogové okno - nevyplnění popisu práce.....	31
Obr. 18: Dialogové okno - úprava profilu.....	32
Obr. 19: Statistiky - počet ověřených firem.....	33
Obr. 20: Rozdíl formulářů - přidat firmu (vlevo) / přidat adresu firmy (vpravo).....	35
Obr. 21: Dialogové okno - potvrzení smazání projektu.....	36
Obr. 22: Profil po přihlášení + výchozí profilová fotografie.....	37

Příloha č. 1

Historie Task Manageru z pohledu P. Mojžíše

„S prvním správcem úkolů jsem se seznámil v roce 2010 v jedné nadnárodní společnosti. Neměl téměř žádné spektrum funkcí, bez vyšších nadstaveb ulehčujících práci, či umožňujících lepší čitelnost textu (např. formátování, zápis pomocí wysiwyg editoru, dynamické přetahování kontaktů, možnost psaní rychlých zpráv, aj.). Než předávat úkoly k doplnění informací, bylo často jednodušší kolegům zavolat anebo se jít zeptat.

V té době jsem zároveň aktivně podnikal a v rámci rozvoje vlastní firmy a integrace jednotlivých nástrojů jsme začali uvažovat o interním SW pro správu projektů, zákazníků a úkolů.

V roce 2011 jsem realizoval miniaplikaci pro správu zákazníků, plateb a služeb ve společnosti Convex systems. Jednalo se o vývoj aplikace na míru, u které jsem využil některé myšlenky a prvky v rámci předchozí analýzy různých úkolovníků.

V roce 2012 se rodina námi nabízených produktů rozrostla z e-shopů a zakázkové grafiky o aplikaci pro správu webů a dokumentů a o aplikaci e-note pro psaní poznámek a datové úložiště, které chystáme nasadit masově v roce 2013. Kvalitní support zákazníkům je jedním z cílů naší služby, proto vznikl doprovodný projekt „TaskManager“, který měl rodinu produktů doplnit a rozšířit formou customizovatelného Project CRM systému.

Během začátku roku 2012 vznikal datový model, později i jádro aplikace. Nebyly však kapacity pro vytvoření uživatelského rozhraní a otestování komplexních funkcionalit. V této době jsem se seznámil s Honzou, který se v rámci studentských projektů velice osvědčil, a nabídl jsem mu k realizaci bakalářské práce právě výše zmíněné UI s otestováním funkcí navázanými na jádro CRM. “

Příloha č. 2

Hodnocení práce J. Wankeho:

„Honza od počátku projevil o práci samotnou velký zájem. Projekt mu přišel smysluplný s možností rozšířit a prohloubit své znalosti objektového programování, práce s novými technologiemi a jejich vzájemnou integrací. K tomuto předpokladu jsem mu mohl nabídnout své zkušenosti z předešlých projektů nejen po technické stránce. Na tomto základě vzniklo zadání bakalářské práce, kterou za TUL vede paní Ing. Alena Gregová, které bych chtěl tímto zároveň poděkovat za přijetí, schválení a příjemnou spolupráci na projektu.

U projektu jsme si stanovili harmonogram a rozdělili práci na etapy a jednotlivé úkoly.

- *Seznámení s technologiemi (JavaScript, jQuery, Ajax, Object PHP, CSS, SQL, aj.)*
- *Seznámení s jádrem CRM, nabízenými funkcemi a jejich vzájemnou integrací*
- *Kompletní návrh UI, propojení s funkcemi CRM a zobrazování výstupu pomocí grafického UI*
- *Popsání výsledků v rámci bakalářské práce, sumarizace poznatků a zkušeností*

Cílem práce bylo vytvořit UI, zkombinovat dostupné technologie pro jeho tvorbu a poté vytvořit uživatelský manuál, jak celý tento systém obsluhovat.

Spolupráce byla skvělá po všech stránkách a pomohla mi více poznat jednotlivé vlastnosti Honzy. Musím především pochválit samostatnost při řešení dílčích i komplexnějších úkolů. Dodržování vytyčených cílů a schopnost přicházet s vlastními myšlenkami a nápady. Výsledkem práce je velice povedené UI, které je svou kvalitou srovnatelné s komerčními projekty podobného rozsahu. Zároveň se zde projevila tvůrčí stránka a smysl pro detail, který je pro uživatelské rozhraní klíčový. Právě smysl pro detail s projevem neuvěřitelné houževnatosti byl někdy příčinou vyšší časové zátěže některých dílčích úkolů. Ve výsledku však nijak nebránily a nakonec Honza dovedl správce úkolů do 100 % stavu.

Jedním z hlavních úkolů, které tento projekt synergicky doprovázely, bylo ověřit si spolupráci na složitějším projektu, schopnosti a týmového ducha. Všechny tyto aspekty Honza splňuje. Kromě jiného byla tato práce vstupním pohovorem a možností, jak předat zkušenosti a část know-how pro urychlení další spolupráce. I tento aspekt Honza splnil na 110% a těším se na naši další spolupráci.“

Petr Mojžíš, Dis.

V Liberci dne 31. 12. 2012.

Příloha č. 3

```
public static function formPridatProjekt($idContact) {
    $address = cCon2Adr_Team2Project::getAdr2ConList($idContact);
    $employee = cCon2Adr_Team2Project::getCon2AdrList($address[0]->getAdrId());

    $empIDs = array();
    for($i = 0; $i<count($employee); $i++){
        $empIDs[$i] = $employee[$i]->getConId();
    }

    $employyyUnique = array_unique($empIDs);

    $allEmployees = cContact::getContactList();
    $form = '
        <form action="'.$_SERVER['PHP_SELF'].'" method="post" enctype="multipart/form-
data" id="multipage" onsubmit="return validate();">
        <fieldset>
            <div class="info">Info o projektu</div>
            <div id="clear"></div>

            <label for="nazevProjektu">Název projektu:</label>
            <input type="input" name="nazevProjektu" id="nazevProjektu" class="styledBG" onkeyup=
="PRJnazev(this.value)" title="Zadejte název projektu">

            <label for="popisProjektu">Popis projektu:</label>
            <textarea cols="" rows="" name="popisProjektu" id="popisProjektu" onkeyup="PRJpopi
s(this.value)" title="Zadejte popis projektu"></textarea>

            <label for="cenaProjektu">Cena projektu:</label>
            <input type="input" name="cenaProjektu" id="cenaProjektu" class="styledBG" onkeyup="
PRJcena(this.value)" title="Zadejte cenu projektu v Kč">

            <div id="clear"></div>
            <label for="verejny">Veřejný:</label>
            <label class="myCheckbox">
                <input type="checkbox" name="verejny" id="verejny" value="1">
            <div></div>
            </label>

            <label for="statusProjektu" id="statusProjektuLabel">Status projektu:</label>
            <select id="statusProjektu" name="statusProjektu" class="styledSelect">
                <option value="0">pre-sale</option>
                <option value="1">prepare</option>
                <option value="2">open</option>
                <option value="3">run</option>
                <option value="4">to finish</option>
                <option value="5">after deadline</option>
                <option value="6">close</option>
                <option value="7">stopped</option>
            </select>
```

```

<label for="priraditProjekt" id="priraditProjektLabel">Přiřadit zaměstnanci:</label
>

<select id="priraditProjekt" name="priraditProjekt" class="styledSelect">';

    for($i = 0; $i < count($allEmployees); $i++) {
        for($j = 0; $j < count($employyUnique); $j++) {
            if ($employyUnique[$j] == $allEmployees[$i]->getId()) {
                $form .= '<option value="'. $allEmployees[$i]->getId().'">' . $allEmployees[$i]-
>getFirstName(). ' ' . $allEmployees[$i]->getLastName(). '</option>';
            }
        }
    }

    $form .= '</select>
    <input type="hidden" value="'. $address[0]->getAdrId().'" name="addressId">
</fieldset>
<div id="formSeparator"></div>
    <input type="submit" value="Odeslat" name="odeslatPridatProjekt" id="odeslatPrid
atProjekt" >
    </form>
';
    return $form;
}

```

Příloha č. 4

Kopie smlouvy o ochraně obchodního tajemství a důvěrných informací.

